

## Практическое занятие № 17

**Цели практического занятия:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с использованием GUI Tkinter в IDE PyCharm Community, изучить возможности модуля OS.

### Вариант № 9

#### Постановка задачи 1 (вторая задачи из трёх):

Разработать программу с применением пакета tk, взяв в качестве условия одну любую задачу из ПЗ №№ 2 – 9. (условие взято из ПЗ2)

#### Код программы:

```
import tkinter as tk

def calculate_segments():
    # получаем данные от пользователя из полей ввода
    A = int(entry_A.get())
    B = int(entry_B.get())

    # выполняем расчет и выводим результат в поле для вывода результата
    if A > B:
        count = A // B
        result_var.set("Количество отрезков B на отрезке A: " + str(count))
    else:
        result_var.set("Введите значение удовлетворяющие условию A>B")

# окно
root = tk.Tk()
root.title("Расчет отрезков")
root.geometry("300x150")

#поля для ввода данных
entry_A_label = tk.Label(root, text="Введите длину отрезка A:")
entry_A_label.grid(row=0, column=0)
entry_A = tk.Entry(root)
entry_A.grid(row=0, column=1)

entry_B_label = tk.Label(root, text="Введите длину отрезка B:")
entry_B_label.grid(row=1, column=0)
entry_B = tk.Entry(root)
entry_B.grid(row=1, column=1)

# кнопка
calculate_button = tk.Button(root, text="Расчет", command=calculate_segments)
calculate_button.grid(row=2, column=0, columnspan=2)

#вывод
result_var = tk.StringVar()
result_label = tk.Label(root, textvariable=result_var)
```

```
result_label.grid(row=3, column=0, columnspan=2)

root.mainloop()
```

#### Протокол выполнения программы:

1. Запускаем программу.
2. Вводим в поле "Введите длину отрезка А" значение длины отрезка А.
3. Вводим в поле "Введите длину отрезка В" значение длины отрезка В.
4. Нажимаем кнопку "Расчет".
5. Программа выполняет расчет количества отрезков В, которые можно разместить на отрезке А.
6. Если  $A > B$ , то программа выводит в поле "Результат" сообщение "Количество отрезков В на отрезке А: " и количество отрезков.
7. Если  $A \leq B$ , то программа выводит в поле "Результат" сообщение "Введите значение удовлетворяющие условию  $A > B$ ".
8. Завершение работы программы.

#### Постановка задачи 2 (третья задача из трёх):

Задание предполагает, что у студента есть проект с практическими работами (№№ 2-13), оформленный согласно требованиям. Все задания выполняются с использованием модуля OS:

перейдите в каталог PZ11. Выведите список всех файлов в этом каталоге. Имена вложенных подкаталогов выводить не нужно.

перейти в корень проекта, создать папку с именем test. В ней создать еще одну папку test1. В папку test переместить два файла из ПЗ6, а в папку test1 - один файл из ПЗ7. Файл из ПЗ7 переименовать в test.txt. Вывести в консоль информацию о размере файлов в папке test.

перейти в папку с PZ11, найти там файл с самым коротким именем, имя вывести в консоль. Использовать функцию `basename()` (`os.path.basename()`).

перейти в любую папку где есть отчет в формате .pdf и «запустите» файл в привязанной к нему программе. Использовать функцию `os.startfile()`.

удалить файл test.txt

#### Код программы:

```
# перейдите в каталог PZ11. Выведите список всех файлов в этом каталоге. Имена
# вложенных подкаталогов выводить не нужно.
```

```

import os

print("Пункт 1")
print('-' * 100)
os.chdir("../PZ11")
files = []

for obj in os.listdir():

    if os.path.isfile(obj):

        files.append(obj)

print(files)

# } перейти в корень проекта, создать папку с именем test. В ней создать еще одну
# папку
# test1. В папку test переместить два файла из ПЗ6, а в папку test1 - один файл
# из ПЗ7.
# Файл из ПЗ7 переименовать в test.txt. Вывести в консоль информацию о размере
# файлов в папке test.

print("Пункт 2")
print('-' * 100)

os.chdir("..")
os.mkdir("test") # создаем папку test
os.mkdir("test/test1") # создаем папку test1 внутри папки test

# копируем файлы из ПЗ6 в папку test
with open("./PZ6/pz6_ivahnenko.pdf", "rb") as src_file:
    with open("test/pz6_ivahnenko.pdf", "wb") as dst_file:
        dst_file.write(src_file.read())

with open("./PZ6/pz6_ivahnenko.py", "r", encoding="utf-8") as src_file:
    with open("test/pz6_ivahnenko.py", "w", encoding="utf-8") as dst_file:
        dst_file.write(src_file.read())

# копируем файл из ПЗ7 в папку test1 и переименовываем его в test.txt
with open("./PZ7/pz7_ivahnenko.py", "r", encoding="utf-8") as src_file:
    with open("test/test1/test.txt", "w", encoding="utf-8") as dst_file:
        dst_file.write(src_file.read())

# получаем размеры всех файлов в папке test
sizes = []
for file in os.listdir("test"):

```

```

    if os.path.isfile(os.path.join("test", file)):
        sizes.append(os.path.getsize(os.path.join("test", file)))

print(sizes)

# перейти в папку с PZ11, найти там файл с самым коротким именем, имя вывести в
# консоль. Использовать функцию basename () (os.path.basename()).

print("Пункт 3")
print('-' * 100)

import os

os.chdir("../PZ11")

shortest_filename = ""
for filename in os.listdir():
    if len(filename) < len(shortest_filename) or shortest_filename == "":
        shortest_filename = filename

print(os.path.basename(shortest_filename))

print("Пункт 4")
print('-' * 100)

pdf_folder = '../PZ2'

pdf_filename = 'ПР2 ивахненко.pdf'

pdf_path = os.path.join(pdf_folder, pdf_filename)

if os.path.isfile(pdf_path):
    # Открытие функции os.startfile()
    os.startfile(pdf_path)
else:
    print("такого файла нет")

print("Пункт 5")
print('-' * 100)

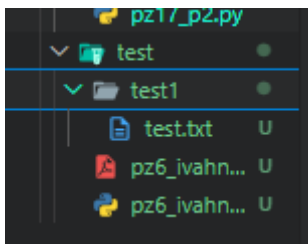
```

```
os.chdir = '../test/test1'

file_path = os.path.join(os.chdir, 'test.txt')

if os.path.isfile(file_path):
    # удаляем файл
    os.remove(file_path)
    print('Файл успешно удален.')
else:
    print('Файл не найден.')
```

#### Результат работы программы:



Пункт 1

-----

['poem.txt', 'pz11\_ivahnenko.pdf', 'pz11\_ivahnenko.py', 'text18-9.txt']

Пункт 2

-----

[437438, 6006]

Пункт 3

#### Протокол выполнения программы:

1. Переход в каталог PZ11. Вывод списка всех файлов в этом каталоге. Имена вложенных подкаталогов выводить не нужно.

- \* Изменение текущего каталога на PZ11 с помощью os.chdir().
- \* Создание пустого списка files для хранения имен файлов.
- \* Перебор всех объектов в текущем каталоге с помощью os.listdir().
- \* Проверка каждого объекта на то, является ли он файлом, с помощью os.path.isfile().
- \* Добавление имен файлов в список files.
- \* Вывод списка files на экран.

2. Переход в корень проекта, создание папки с именем test. В ней создать еще одну папку test1. В папку test переместить два файла из ПЗ6, а в папку test1 - один файл из ПЗ7. Файл из ПЗ7 переименовать в test.txt. Вывести в консоль информацию о размере файлов в папке test.

- \* Изменение текущего каталога на корень проекта с помощью `os.chdir()`.
- \* Создание папки test с помощью `os.mkdir()`.
- \* Создание папки test1 внутри папки test с помощью `os.mkdir()`.
- \* Копирование файлов из ПЗ6 в папку test с помощью `open()` и `os.path.join()`.
- \* Копирование файла из ПЗ7 в папку test1 и переименование его в test.txt с помощью `open()` и `os.path.join()`.
- \* Получение размеров всех файлов в папке test с помощью `os.listdir()`, `os.path.isfile()` и `os.path.getsize()`.
- \* Создание списка sizes для хранения размеров файлов.
- \* Вывод списка sizes на экран.

3. Переход в папку с PZ11, найти там файл с самым коротким именем, имя вывести в консоль. Использовать функцию `basename()` (`os.path.basename()`).

- \* Изменение текущего каталога на PZ11 с помощью `os.chdir()`.
- \* Создание пустой строки `shortest_filename` для хранения имени файла с самым коротким именем.
- \* Перебор всех файлов в текущем каталоге с помощью `os.listdir()`.
- \* Сравнение длины текущего имени файла с длиной `shortest_filename`.
- \* Изменение значения `shortest_filename` на текущее имя файла, если оно короче.
- \* Вывод значения `shortest_filename` на экран с помощью `os.path.basename()`.

4. Открыть файл с именем 'ПР2 ивахненко.pdf', который находится в папке PZ2. Использовать функцию `os.startfile()`.

- \* Создание переменной `pdf_folder` для хранения пути к папке PZ2.
- \* Создание переменной `pdf_filename` для хранения имени файла.
- \* Создание переменной `pdf_path` для хранения полного пути к файлу с помощью `os.path.join()`.
- \* Проверка существования файла с помощью `os.path.isfile()`.
- \* Открытие файла с помощью `os.startfile()` при условии, что файл существует.
- \* Вывод сообщения "такого файла нет" на экран, если файл не существует.

5. Переход в папку test/test1, удалить файл test.txt.

- \* Изменение текущего каталога на test/test1 с помощью `os.chdir`.
- \* Создание переменной `file_path`, которая содержит путь к файлу `test.txt` с помощью `os.path.join()`.
- \* Проверка существования файла с помощью `os.path.isfile()`.
- \* Удаление файла с помощью `os.remove()`

**Вывод:**

Я закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ с использованием GUI Tkinter в IDE PyCharm Community, изучил возможности модуля OS.